Calculating the Fickian diffusivity for a lattice–based random walk with agents and obstacles of different shapes and sizes

Adam J Ellery

School of Mathematical Sciences, Queensland University of Technology (QUT), Brisbane, Australia.

Ruth E Baker

Mathematical Institute, University of Oxford,

Radcliffe Observatory Quarter, Woodstock Road, Oxford, UK.

Matthew J Simpson

School of Mathematical Sciences, QUT, Brisbane, Australia.

Abstract

Random walk models are often used to interpret experimental observations of the motion of biological cells and molecules. A key aim in applying a random walk model to mimic an *in vitro* experiment is to estimate the Fickian diffusivity (or Fickian diffusion coefficient), D. However, many *in vivo* experiments are complicated by the fact that the motion of cells and molecules is hindered by the presence of obstacles. Crowded transport processes have been modeled using repeated stochastic simulations in which a motile agent undergoes a random walk on a lattice that is populated by immobile obstacles. Early studies considered the most straightforward case in which the motile agent and the obstacles are the same size. More recent studies considered stochastic random walk simulations describing the motion of an agent through an environment populated by obstacles of different shapes and sizes. Here, we build on previous simulation studies by analyzing a general class of lattice–based random walk models with agents and obstacles of various shapes and sizes. Our analysis provides exact calculations of the Fickian diffusivity, allowing us to draw conclusions about the role of the size, shape and density of the obstacles, as well as examining the role of the size and shape of the motile agent. Since our analysis is exact, we calculate D directly without the need for random walk simulations. In summary, we find that the shape, size and density of obstacles has a major influence on the exact Fickian diffusivity. Furthermore, our results indicate that the difference in diffusivity for symmetric and asymmetric obstacles is significant.

I. INTRODUCTION

Random walk models are routinely used to mimic both *in vitro* and *in vivo* experimental observations describing the motion of cells and biomolecules [1–5]. Often the aim of applying a random walk model to mimic *in vitro* observations is to provide an estimate of the diffusivity, D [6–8]. Unlike simpler *in vitro* experiments, *in vivo* experiments are complicated by the fact that the cells and biomolecules move through an environment populated by various types of obstacles that, owing to crowding effects, hinder their motion [9–12].

Early analysis of random walk models describing motion through crowded environments involved repeated stochastic simulations of the random walk of an agent on a lattice, where the lattice is populated by immobile obstacles [13, 14]. In these early models both the size of the agent and the obstacles are equal to the lattice spacing [13, 14]. Analyzing the temporal evolution of the mean squared displacement (MSD) of the motile agent indicates that, after a sufficient period of time, the MSD grows in proportion to time. Measuring the constant of proportionality from repeated computer simulations provides a means of estimating the long–time Fickian diffusivity, D [13–16]. Using this approach, it is possible to demonstrate that D decreases as the density of obstacles increases, as we might anticipate [13–16]. Other avenues of investigation involved the use of perturbation theory [17], percolation theory [18–20] and the eigenvalue spectrum [21]. More recent studies attempt to provide more realistic descriptions of cell and biomolecule motion by considering the motion of agents through environments that are populated by obstacles of varying shapes and sizes [22–24]. Similar to earlier models, these more detailed simulation studies show that, after a sufficient period of time, the MSD increases in proportion to time, and this observation can also be used to estimate the long–time Fickian diffusivity, D [22, 23]. In addition to showing that D depends on the density of obstacles, these studies suggest that D also depends on the size and shape of the obstacles [22, 23].

Instead of relying on repeated stochastic random walk simulations, here we present a method that enables us to exactly calculate the long–time Fickian diffusivity, D, for a lattice–based random walk in which we vary the shape and size of both the motile agent and the immobile obstacles. We achieve this by generalizing an approach previously described by Mercier and Slater [26–28]. The algorithm is described in Section II, where we also confirm the accuracy of the approach by comparing exact calculations of D with estimates obtained using standard simulation methods. In Section III A we systematically vary the size of the agent, the size of the obstacles and the density of the obstacles to examine how these details affect D for a two–dimensional random walk in which both the agents and obstacles are symmetric in shape. In Section III B we consider a suite of two–dimensional scenarios in which either the agent or the obstacles are asymmetric. We also demonstrate how our approach can be implemented for three–dimensional problems in Section III C where we systematically vary the

size, shape and density of symmetric obstacles in three–dimensions. In Section III D we calulate D for obstacle densities up to 50%. Finally, in Section IV, we discuss the implications of our results.

For symmetric agents and obstacles of the same size, we show that D decreases with the density of obstacles. This is an expected result that has been observed previously in simulation studies, however we also make further observations that have not been identified previously. For a given obstacle density, we find that D is most reduced for larger motile agents, relative to smaller motile agents, provided that the obstacles are symmetric. However, the opposite can be true when we consider the motion of agents in an environment populated by asymmetric obstacles. Furthermore, we find that D is sensitive to the size of the agents and obstacles relative to the size of the lattice spacing. In summary, our results indicate that the value of D is strongly linked to the details of the shape, size, symmetry and density of the obstacles. Therefore, caution ought to be exercised when D is estimated without a detailed characterization of the obstacle shape, size and density.

II. THEORY

We now consider a two–dimensional square lattice of dimension $X \times Y$, with unit lattice spacing $\Delta = 1$. Sites are indexed (i, j) so that each site has location $(x, y) = (i, j)$, with $0 \leq x \leq (X - 1)$ and $0 \leq y \leq (Y - 1)$. We also consider a three–dimensional square lattice of dimension $X \times Y \times Z$, with unit lattice spacing $\Delta = 1$. Sites are indexed (i, j, k) so that each site has location $(x, y, z) = (i, j, k)$, with $0 \leq x \leq (X - 1)$, $0 \leq y \leq (Y - 1)$ and $0 \leq z \leq (Z - 1)$. Since we always work with unit lattice spacing and unit rate constants, all calculations of diffusivity are dimensionless. These calculations can be redimensioned by re–scaling with an appropriate dimensional lattice spacing, Δ, if required.

We represent various crowding environments by randomly populating a lattice with immobile obstacles to a specified spatially uniform density, $\phi = m_0\nu/m$, where m_0 is the number of obstacles, ν is the number of sites per obstacle, and m is the total number of lattice sites. For the two–dimensional lattice we consider the five different types of agents and obstacles depicted in Figure 1, namely: (i) a square obstacle of area 1 (1×1, $\nu = 1$); (ii) a square obstacle of area 4 (2×2, $\nu = 4$); (iii) an asymmetric obstacle of area 2, consisting of two adjacent lattice sites (1×2, $\nu = 2$); (iv) an asymmetric obstacle of area 3 in an L–shaped arrangement ($\nu = 3$); and (v) an asymmetric obstacle of area 3 consisting of three lattice sites in a row (1×3, $V = 3$). When placing asymmetric obstacles on the lattice we always take care to randomly orient the obstacles so that, on average, there is no preferred direction of alignment. For the three–dimensional lattice we consider two types of obstacles that are also depicted in Figure 1, namely: (vi) a cubic obstacle of volume 1 ($1 \times 1 \times 1$, $\nu = 1$); and (vii) a cubic obstacle of volume 8 ($2 \times 2 \times 2$, $\nu = 8$). Although our results focus on these seven specific obstacle and agent shapes, the exact procedure for calculating D can be applied to other obstacle and agent

shapes by extending our analysis in an obvious way. Throughout this study we always ensure that ϕ is well below the relevant percolation threshold [29]. In the first part of the study we focus on modest obstacle densities, $0 < \phi \leq 0.15$. In the second part of the study we provide additional results for higher obstacle densities, $\phi \leq 0.50$.

To provide a check on the accuracy of our exact calculations of D we perform stochastic random walk simulations with the aim of showing that the exact calculations match estimates of D from a simulation approach. To initiate the random walk simulations we place a single motile agent at a randomly chosen location on the lattice, being careful to ensure that the agent does not overlap with any obstacles. The shape of the agent is chosen from one of the types depicted in Figure 1. The agent undergoes an unbiased nearest neighbor random walk with unit step length ($\Delta = 1$), periodic boundary conditions and an exclusion condition [32]. The exclusion condition means that any potential motility event that would lead to part of the agent occupying any site that is already occupied by an obstacle is aborted [32]. Our discrete model is similar to a blind-ant random walk [18]. We use the Gillespie algorithm [31] to advance the simulation through time until we reach a pre–specified termination time, T. In all situations the rate at which the motile agent attempts to undergo a motility event is set to unity. Since the random walk simulations are stochastic, we always average our results over a large ensemble of identically prepared realizations. As each random walk simulation proceeds, we track the position of the agent and record the agent trajectory, from which we can calculate the MSD.

After performing a large number of identically prepared realizations of each random walk, we make the standard assumption that the average MSD data follows a power law [13–16]

$$\left\langle r^2 \right\rangle = (2dD)t^{\alpha}, \tag{1}$$

where $r^2 = x^2 + y^2 + z^2$ is the dimensionless displacement squared, t is dimensionless time, d is the dimension of the system, α is an exponent, $0 < D \leq 1/(2d)$ denotes the dimensionless diffusivity and $\langle \cdot \rangle$ denotes an average over a large ensemble. Fickian diffusion is associated with $\alpha = 1$ [1]. To estimate D from this data we rewrite Eq. (1) as

$$\log_{10}\left(\frac{\left\langle r^2 \right\rangle}{t}\right) = \log_{10}\left(2dD\right) + \left(\alpha - 1\right)\log_{10}\left(t\right). \tag{2}$$

Equation (2) suggests that if the data follows this power law, when we plot $\log_{10}\left(\left\langle r^2 \right\rangle / t\right)$ as a function of $\log_{10}(t)$ we should observe a straight line with slope $\alpha - 1$. It is well–known that for MSD data generated in this way, a plot of $\log_{10}\left(\left\langle r^2 \right\rangle / t\right)$ as a function of $\log_{10}(t)$ asymptotes to a horizontal line [13–16]. This indicates that, after a sufficient amount of time we have $\alpha = 1$ and we can estimate the Fickian diffusion coefficient by estimating the intercept of the horizontal asymptote of the plot of $\log_{10}\left(\left\langle r^2 \right\rangle / t\right)$ as a function of $\log_{10}(t)$ [13–16].

Instead of recording the square of the total displacement, $r^2 = x^2 + y^2 + z^2$, an alternative approach is to record the temporal evolution of the square of the displacement in each component direction

and apply a similar technique. This approach would allow us to estimate the diffusion coefficient in each direction. For example, treating the x, y and z components of the MSD data in this way can be used to provide estimates of D_x, D_y and D_z, respectively, allowing us to explore whether there are any differences in the diffusivity in each direction. Here, $0 < D_x \leq 1/d$, $0 < D_y \leq 1/d$ and $0 < D_z \leq 1/d$, where d is the dimension of the problem. To implement this approach we plot $\log_{10}\left(\langle x^2 \rangle / t\right)$, $\log_{10}\left(\langle y^2 \rangle / t\right)$ or $\log_{10}\left(\langle z^2 \rangle / t\right)$ as a function of $\log_{10}(t)$, and we estimate D_x, D_y and D_z from measuring the long–time intercept of these curves, which asymptote to $\log_{10}\left(2D_x\right)$, $\log_{10}\left(2D_y\right)$ and $\log_{10}\left(2D_z\right)$, respectively.

Using this approach we plot the MSD data, focusing at first on the x–component of the trajectory data, for several different combinations of different sized agents and obstacles, in both two and three dimensions, in Figure 2. Results in Figure 2(a) show plots of $\log_{10}\left(\langle x^2 \rangle / t\right)$ as a function of $\log_{10}(t)$ for both 1×1 and 2×2 shaped agents undergoing a random walk on a two–dimensional lattice that is randomly populated by either 1×1 obstacles or 2×2 obstacles. Results in Figure 2(b) show plots of $\log_{10}\left(\langle x^2 \rangle / t\right)$ as a function of $\log_{10}(t)$ for both 1×2 and 1×1 agents and obstacles as well as results for both 1×1 and L–shaped agents and obstacles. Similar plots for three–dimensional random walk simulations are shown in Figure 2(c) for both a $1 \times 1 \times 1$ and $2 \times 2 \times 2$ agent moving on a three–dimensional lattice populated with either $1 \times 1 \times 1$ or $2 \times 2 \times 2$ obstacles. Regardless of the dimension of the problem, or the combination of agent and obstacle shape and size, each plot in Figure 2 shows that the MSD curve eventually asymptotes to a horizontal line confirming that we eventually observe Fickian diffusion. Since these curves asymptote to $\log_{10}\left(2D_x\right)$, as indicated by Eq. (2), we can estimate D_x from these data by measuring the intercept of the horizontal asymptote. This is a standard approach that has been used in many previous studies [13–16].

Results in Figure 2 suggest that, in both two and three dimensions, smaller obstacles are more effective at reducing the diffusivity than larger obstacles at the same density. For a two–dimensional random walk with symmetric agents and obstacles, as shown in Figure 2(a), we find that a lattice populated with small obstacles is most effective at reducing the diffusivity, and that this effect is more pronounced for larger agents. Results for a two–dimensional random walk with asymmetric agents and obstacles, as shown in Figure 2(b), also suggest that small obstacles are more effective at reducing the diffusivity and that this reduction is more pronounced if the agent is asymmetric than if it is symmetric. Comparing the data in Figure 2(a)–(b) suggests that the reduction in diffusivity is greater for a 2×2 agent than for an asymmetric 1×2 agent. Data in Figure 2(c) indicate that similar trends apply in three dimensions.

In practice, trajectory data from stochastic random walk simulations is very noisy [25] and to make sense of these data it is necessary to average over a very large ensemble of identically prepared realizations to obtain accurate estimates of the long–time diffusivity. In some cases, calculating the

diffusivity directly from stochastic random walk simulations can be overwhelmingly computationally expensive. This observation motivates us to apply an alternative, exact approach.

A. Algorithm and worked example

We now calculate the diffusivity using the method proposed by Mercier and Slater [26–28]. The Mercier–Slater method involves applying a vanishingly small bias, $\epsilon > 0$, to the random walk making it more probable for the agent to attempt to move in a particular direction. The bias can be applied separately in either the x, y or z directions, and is therefore applicable to random walks in both two and three dimensions. If the bias is applied in the x–direction, we calculate the exact diffusivity in that direction using the Nernst–Einstein relationship (a special case of the fluctuation-dissipation theorem [30]),

$$\mathcal{D}_x = \frac{D_x}{D0_x} = \lim_{\epsilon \to 0} \frac{\mu(\epsilon)}{\mu_0}, \tag{3}$$

where $0 < D_x \leq 1/d$ is the dimensionless diffusivity in the x–direction, $D0_x = 1/d$ is the dimensionless diffusivity in the x–direction when there are no obstacles present, $\mu(\epsilon)$ is the probability of the agent moving in the x–direction under the action of the bias and μ_0 is the probability of the agent moving in the x–direction without the influence of the bias. $\mathcal{D}_x$ denotes the ratio of D_x to $D0_x$ so that $0 < \mathcal{D}_x \leq 1$. Applying the bias in the y or z–directions can be used to calculate $\mathcal{D}_y$ or $\mathcal{D}_z$, respectively.

Throughout this work we always apply the Mercier–Slater method by applying the bias in the positive x, y or z directions. We note that applying the bias in the negative x, y or z direction does not change our results since the sign of ϵ does not affect the value of the limit in Eq. (3). Since $\mu(\epsilon)$ and μ_0 do not vary with time, we can calculate them by considering properties of the long–time expected motion of the agent on the lattice. Equation (3) thus permits an exact calculation of the long–time Fickian diffusivity, and it can be applied to each combination of agent and obstacle shapes and sizes in Figure 2.

We calculate $\mu(\epsilon)$ from the mean agent velocity at each lattice site. Let $\mathbf{v}$ be a vector whose q^{th} element, $v(q)$, denotes the mean agent velocity at site q. If we consider the x direction, the mean agent velocity is given by

$$v(q) = p_+ L_+ - p_- L_-, \tag{4}$$

where $p_\pm$ are the probabilities of movement in the positive and negative x–directions, respectively, and $L_\pm = 1$ if the relevant target site is vacant and zero if it is occupied.

The probability of finding the agent at any particular lattice site in the long–time limit is given by solving the eigenvalue problem

$$\mathbf{T}\,\mathbf{n} = \mathbf{n},$$

where $\mathbf{T}$ represents a transition matrix in which the element of the a^{th} row and b^{th} column, T_{ab}, is the probability that an agent located at site b will step to site a in the next time step, and $\mathbf{n}$ is a vector whose q^{th} element, $n(q)$, denotes the probability that an agent is located at site q in the long–time limit. Generally, we normalize $\mathbf{n}$ to ensure that $\sum_q n(q) = 1$, and we do this by calculating the unit vector, $\hat{\mathbf{n}}$. The mean (global) velocity is given by $\mathbf{v} \cdot \hat{\mathbf{n}}$, and from these quantities we can derive

$$\mu(\epsilon) = \frac{\mathbf{v} \cdot \hat{\mathbf{n}}}{\epsilon}, \tag{5}$$

from which we can recover the diffusivity using Eq. (3).

In summary, applying the Mercier–Slater algorithm involves the following steps:

1. for a given obstacle configuration, construct the transition matrix, $\mathbf{T}$;
2. solve $\mathbf{T}\,\mathbf{n} = \mathbf{n}$;
3. calculate $\hat{\mathbf{n}}$;
4. calculate the mean velocity, $\mathbf{v}$, at each lattice site;
5. calculate $\mu(\epsilon)$ using Eq. (5);
6. calculate $\mathcal{D}_x$ using Eq. (3).

To save a mathematical step we take advantage of the fact that the row rank of $(\mathbf{T} - \mathbf{I})$ is less than or equal to the number of rows of $\mathbf{T}$, and is only equal in the special case in which $\mathbf{T} = \mathbf{I}$. Accordingly, we introduce the normalization condition by rewriting $\mathbf{T}\,\mathbf{n} = \mathbf{n}$ as $(\mathbf{T} - \mathbf{I})\,\mathbf{n} = \mathbf{0}$, where $\mathbf{I}$ is an identity matrix of appropriate size, and we replace a row of the new coefficient matrix, $(\mathbf{T} - \mathbf{I})$, with ones. This gives $\mathbf{A}\,\mathbf{n} = \mathbf{b}$ where $\mathbf{A}$ is the modified coefficient matrix and $\mathbf{b} = [0, 0, ..., 1]^{\text{T}}$, which enforces the normalization condition.

A further simplification can also be implemented. If $\mathbf{v}_\epsilon$ and $\mathbf{v}_I$ denote the ϵ dependent and ϵ independent components of $\mathbf{v}$, respectively, and $\mathbf{n}_\epsilon$ and $\mathbf{n}_I$ denote the ϵ dependent and ϵ independent components of $\hat{\mathbf{n}}$, respectively, then Eq. (3) and Eq. (5) can be re–expressed in the simpler form [26]

$$\mathcal{D} = \mathbf{v}_\epsilon \cdot \mathbf{n}_I + \mathbf{v}_I \cdot \mathbf{n}_\epsilon. \tag{6}$$

To solve for $\mathbf{n}_\epsilon$ and $\mathbf{n}_I$ we separate $\mathbf{A}$ into its ϵ dependent and ϵ independent components (so that $\mathbf{A} = \mathbf{A}_I + \epsilon\,\mathbf{A}_\epsilon$), and note that $\mathbf{n}_I$ and $\mathbf{n}_\epsilon$ satisfy [26]

$$\mathbf{A}_I\,\mathbf{n}_I = \mathbf{b}, \tag{7}$$

$$\mathbf{A}_I\,\mathbf{n}_\epsilon = -\mathbf{A}_\epsilon\,\mathbf{n}_I. \tag{8}$$

There are a number of special cases in which the algorithm leads to interesting results. Here we will explicitly discuss two of these special cases:

(i) It is possible that the obstacles may be placed in a way that prevents the agent from moving. In Figure (3)(a) we show an example of this special case. Here, we have $\mathbf{T} = \mathbf{I}$ and the algorithm gives

$$\mathbf{n} = \left(\frac{1}{N}, \frac{1}{N}, ..., \frac{1}{N}\right)^{\mathrm{T}},$$
$$\mathbf{v} = \mathbf{0},$$

where N is the total number of vacant lattice sites. For this system, Eq. (6) gives $\mathcal{D} = 0$, as expected.

(ii) In the special case in which the lattice contains closed and independent regions, the agent will remain trapped in the starting region for all time. We show two example lattices, in Figure (3)(b)–(c), in which obstacles are laid down in a manner that separates the available lattice sites into two independent regions, each of which has an independent set of probability equations. There are two possible ways of dealing with this particular lattice. First, if we take the usual approach and consider the entire lattice we find that $\mathbf{T}$ is block diagonal and $\mathcal{D} = 0$. Second, if we treat each independent region separately we find that $\mathcal{D} > 0$ in each region.

For all of the results presented in this work, we always consider a random arrangement of obstacles, and so we do not focus on these special cases any further.

To demonstrate the application of the algorithm we present a worked example for the lattice and obstacle configuration shown in Figure 4(a). This is a two–dimensional lattice, of size 4×2, containing a 1×2 obstacle in the lower left portion of the lattice. If we bias the motion in the positive x–direction and consider the case where the agent is initially located at site 1, the probability that the agent will step to site 2 is $(1 + \epsilon)/4$ and the probability that the agent will step to site 4 is $(1 - \epsilon)/4$. The agent will attempt to step in the negative y–direction with probability $1/4$, however this attempted motility event will be aborted because of the obstacle. Similarly, the agent will attempt to step in the positive y direction with probability $1/4$ and this attempted motility event will also be aborted due to the periodic boundary conditions and the location of the obstacle. Therefore, there is a probability of $1/4 + 1/4 = 1/2$ that the agent will remain at site 1. Following a similar process of reasoning for the remaining lattice sites, the transition matrix for this lattice, with a bias in the positive x direction, is given by

$$\mathbf{T} = \begin{pmatrix} \frac{1}{2} & \frac{1+\epsilon}{4} & 0 & \frac{1-\epsilon}{4} & 0 & 0 \\ \frac{1-\epsilon}{4} & \frac{1}{2} & \frac{1+\epsilon}{4} & 0 & 0 & 0 \\ 0 & \frac{1-\epsilon}{4} & 0 & \frac{1+\epsilon}{4} & \frac{1}{2} & 0 \\ \frac{1+\epsilon}{4} & 0 & \frac{1-\epsilon}{4} & 0 & 0 & \frac{1}{2} \\ 0 & 0 & \frac{1}{2} & 0 & \frac{1+\epsilon}{4} & \frac{1+\epsilon}{4} \\ 0 & 0 & 0 & \frac{1}{2} & \frac{1-\epsilon}{4} & \frac{1-\epsilon}{4} \end{pmatrix}.$$

The mean velocity vectors, whose elements are given by Eq. (4), are

$$\mathbf{v}_I = \left(0, 0, 0, 0, -\tfrac{1}{4}, \tfrac{1}{4}\right)^{\mathrm{T}}, \tag{9}$$

$$\mathbf{v}_\epsilon = \left(\tfrac{1}{2}, \tfrac{1}{2}, \tfrac{1}{2}, \tfrac{1}{2}, \tfrac{1}{4}, \tfrac{1}{4}\right)^{\mathrm{T}}. \tag{10}$$

From Eqs. (7)–(8) it follows that

$$\mathbf{n}_I = \left(\tfrac{1}{6}, \tfrac{1}{6}, \tfrac{1}{6}, \tfrac{1}{6}, \tfrac{1}{6}, \tfrac{1}{6}\right)^{\mathrm{T}}, \tag{11}$$

$$\mathbf{n}_\epsilon = \left(-\tfrac{1}{792}, \tfrac{23}{792}, \tfrac{47}{792}, -\tfrac{25}{792}, \tfrac{95}{792}, -\tfrac{73}{792}\right)^{\mathrm{T}}. \tag{12}$$

Combining Eqs. (9)–(12) with Eq. (6) gives $\mathcal{D}_x = 4/11$.

If we bias motion in the positive y–direction, following a similar procedure, the transition matrix for this lattice is

$$\mathbf{T} = \begin{pmatrix} \frac{1}{2} & \frac{1}{4} & 0 & \frac{1}{4} & 0 & 0 \\ \frac{1}{4} & \frac{1}{2} & \frac{1}{4} & 0 & 0 & 0 \\ 0 & \frac{1}{4} & 0 & \frac{1}{4} & \frac{1}{2} & 0 \\ \frac{1}{4} & 0 & \frac{1}{4} & 0 & 0 & \frac{1}{2} \\ 0 & 0 & \frac{1}{2} & 0 & \frac{1}{4} & \frac{1}{4} \\ 0 & 0 & 0 & \frac{1}{2} & \frac{1}{4} & \frac{1}{4} \end{pmatrix}.$$

Note that the coefficients appear to be independent of ϵ owing to a number of algebraic cancellations. For example, the probability of stepping in the positive y direction, from site 3 to site 5, is $(1+\epsilon)/4$. The probability of stepping in the negative y direction, from site 3 to site 5, is $(1-\epsilon)/4$. This means that the net probability of stepping from site 3 to site 5 is $(1+\epsilon)/4 + (1-\epsilon)/4 = 1/2$, which is independent of ϵ. The mean velocity vectors are

$$\mathbf{v}_I = \left(0, 0, 0, 0, 0, 0\right)^{\mathrm{T}}, \tag{13}$$

$$\mathbf{v}_\epsilon = \left(0, 0, \tfrac{1}{2}, \tfrac{1}{2}, \tfrac{1}{2}, \tfrac{1}{2},\right)^{\mathrm{T}}. \tag{14}$$

From Eqs. (7)–(8) it follows that

$$\mathbf{n}_I = \left(\tfrac{1}{6}, \tfrac{1}{6}, \tfrac{1}{6}, \tfrac{1}{6}, \tfrac{1}{6}, \tfrac{1}{6}\right)^{\mathrm{T}}, \tag{15}$$

$$\mathbf{n}_\epsilon = \left(0, 0, 0, 0, 0, 0\right)^{\mathrm{T}}. \tag{16}$$

Combining Eqs. (13)–(16) with Eq. (6) gives $\mathcal{D}_y = 1/3$. Additional worked examples in both two and three dimensions are given in the appendix.

The systems of linear equations that we derive are exact, however we always solve these equations numerically. In general, the transition matrix is large and sparse. Under these conditions Eqs. (7)–(8) can be solved efficiently using the generalised minimal residual method (GMRES) [33]. GMRES is

an iterative method which efficiently solves for the solution to the matrix equation $\mathbf{Ax} = \mathbf{b}$ in the Krylov subspace $K_n = \text{span}\{\mathbf{b}, \mathbf{Ab}, \mathbf{A}^2\mathbf{b}, ..., \mathbf{A}^{n-1}\mathbf{b}\}$ by finding the vector $\mathbf{x}_n$ which minimises the Euclidean norm of the residual $\mathbf{Ax}_n - \mathbf{b}$ to within any given error tolerance [33]. To demonstrate our implementation of this approach we apply it to calculate the diffusivity in the x–direction for each combination of agent and obstacle depicted in Figure 2. Using our estimate of $\mathcal{D}_x$, we plot horizontal lines at $\log_{10}(2D_x)$ on each subfigure in Figure 2. Comparing the averaged MSD data in Figure 2 with the relevant horizontal lines at $\log_{10}(2D_x)$ confirms that there is very good agreement between the exact calculations of D_x and the results from the averaged MSD data since the averaged MSD data asymptotes to the same horizontal line indicated by the exact calculations. Note that all MSD data in Figure 2 corresponds to the x-component of the trajectory data which is why the estimates of diffusivity correspond to the x-component.

III. RESULTS

Now that we have described and verified the algorithm for calculating the diffusivity on a range of small lattices, we apply the algorithm to calculate the diffusivity on larger, more practically sized lattices. We interpret our results by applying the algorithm to a large number of identically-prepared lattices with the same value of ϕ but with randomly arranged obstacles. Since the obstacles are always randomly oriented we always find that $\langle\mathcal{D}_x\rangle = \langle\mathcal{D}_y\rangle = \langle\mathcal{D}_z\rangle$ (results not shown), confirming that there is, on average, no anisotropy. Therefore, for the remainder of this study we only present results for $\langle\mathcal{D}_x\rangle$, and for simplicity we drop the subscript x and the triangular brackets.

A. Two–dimensional random walk with symmetric agents and obstacles

We first consider two–dimensional lattices involving either 1×1 or 2×2 agents moving on a lattice which has been randomly populated with either 1×1 or 2×2 obstacles at density ϕ. We use the Mercier–Slater algorithm to calculate $\mathcal{D}$ for each lattice, and we consider an ensemble of $10,000$ identically prepared lattices. Results are shown as a histogram of $\mathcal{D}$ in Figure 5. The mean estimates of $\mathcal{D}$ are given in Table I.

For each system we consider, our results indicate that $\mathcal{D}$ decreases with increasing obstacle density and that smaller obstacles are more effective at reducing $\mathcal{D}$ than larger obstacles at the same density. For all densities considered, $\mathcal{D}$ is smaller for 1×1 obstacles than for 2×2 obstacles for both 1×1 and 2×2 agents at the same density. We also note that these effects are more pronounced for larger agents than for smaller agents. On average, the diffusivity of a 2×2 agent on a given lattice is smaller than the diffusivity of a 1×1 agents on the same lattice.

We find that, in situations where the agent and obstacles are the same size, the absolute size of the agent and obstacles relative to the size of the lattice spacing has an impact on $\mathcal{D}$. For example, for $\phi = 0.15$, with 1×1 agents and 1×1 obstacles we have $\mathcal{D} = 0.6564$, whereas for a similar situation with 2×2 agents and 2×2 obstacles we have $\mathcal{D} = 0.4849$, corresponding to a reduction in $\mathcal{D}$ by approximately 26%.

A histogram of calculated diffusivities is given in Figure 5. Results in the left column correspond to lattices with $X = Y = 50$ while results in the right column correspond to a larger lattice with $X = Y = 100$. Our results indicate that the mean diffusivity for each combination of agent and obstacle sizes is approximately independent of the size of the lattice. However, the standard deviation of each distribution decreases as the lattice size increases. For example, for a system in which 2×2 agents move through a lattice populated with 2×2 obstacles to density $\phi = 0.15$, we have $\mathcal{D} = 0.4841 \pm 0.0491$ and $\mathcal{D} = 0.4849 \pm 0.0279$ for $X = Y = 50$ and $X = Y = 100$, respectively. This reduction in variability suggests that for a larger lattice fewer realizations are required to estimate $\mathcal{D}$ reliably.

	1×1 Agent		2×2 Agent	
ϕ	1×1 Obstacles	2×2 Obstacles	1×1 Obstacles	2×2 Obstacles
0.05	0.8907 ± 0.0021	0.9192 ± 0.0026	0.6807 ± 0.0133	0.8364 ± 0.0079
0.10	0.7765 ± 0.0045	0.8352 ± 0.0054	0.3573 ± 0.0452	0.6646 ± 0.0176
0.15	0.6564 ± 0.0210	0.7479 ± 0.0081	0.0683 ± 0.0400	0.4849 ± 0.0279

TABLE I. Average diffusivity, $\mathcal{D}$, with the variability indicated by the sample standard deviation, for a system in which a 1×1 agent moves on a lattice randomly populated with 1×1 obstacles. All results correspond to a 100×100 lattice and are calculated from an ensemble of $10,000$ identically prepared realizations.

B. Two–dimensional random walk with asymmetric agents and obstacles

We now consider a two–dimensional system in which either the agent or the obstacles are asymmetric. Specifically, we consider the situation where we have: (i) either a 1×1 agent moving on a lattice populated with 1×2 obstacles or we have a 1×2 agent moving on a lattice populated with 1×1 obstacles; (ii) the situation where we have either a 1×1 agent moving on a lattice populated with L–shaped obstacles or we have a L–shaped agent moving on a lattice populated with 1×1 obstacles; (iii) the situation where we have either a 1×1 agent moving on a lattice populated with 1×3 obstacles or we have a 1×3 agent moving on a lattice populated with 1×1 obstacles. A histogram of the ensemble of the diffusivity results is shown in Figure 6 and the sample mean and variability, given by the sample standard deviation, for each system are shown in Table II.

Our results for the asymmetric agents and obstacles are interesting because some of the trends

we observed in Section III A for symmetric agents and obstacles do not apply. Specifically, while the data in Table II suggest that the smaller 1×1 obstacles are more effective are reducing $\mathcal{D}$ than larger 1×2 obstacles for each density considered. However, we see that larger L–shaped obstacles are more effective at reducing the diffusivity than the smaller 1×1 obstacles for higher densities. 1×3 obstacles are less effective at reducing the diffusivity compared to L–shaped obstacles. This particular result is interesting because it shows that obstacles of the same size can have a different impact on the diffusivity, and we attribute this to the difference in obstacle shape. However, 1×3 agents have a lower diffusivity, on average, than L–shaped agents when they are placed on a lattice populated with 1×1 obstacles. We also observe other differences when we compare the symmetric and asymmetric cases. For example, a 1×1 agent moving on a lattice occupied with L–shaped obstacles to a density of $\phi = 0.15$ is associated with $\mathcal{D} = 0.0898$ whilst an L–shaped agent moving on a lattice occupied with 1×1 obstacles has $\mathcal{D} = 0.1247$, an increase of approximately 40%. Similarly, a 1×1 agent moving on a lattice occupied with 1×3 obstacles has $\mathcal{D} = 0.6560$ whilst a 1×3 agent moving on a lattice occupied with 1×1 obstacles has $\mathcal{D} = 0.0846$, a decrease of approximately 87%. Therefore, unlike the symmetric cases considered in Section III A, increasing the size of asymmetric agents does not necessarily decrease the diffusivity.

	1×2 Agent	1×1 Agent	L–shaped Agent	1×1 Agent	1×3 Agent	1×1 Agent
ϕ	1×1 Obstacles	1×2 Obstacles	1×1 Obstacles	L–shaped Obstacles	1×1 Obstacles	1×3 Obstacles
0.05	0.7240 ± 0.0067	0.7858 ± 0.0073	0.6971 ± 0.0220	0.6999 ± 0.0165	0.5443 ± 0.0077	0.8907 ± 0.0011
0.10	0.4910 ± 0.0216	0.5900 ± 0.0124	0.3985 ± 0.0501	0.3861 ± 0.0391	0.2608 ± 0.0114	0.7763 ± 0.0045
0.15	0.2951 ± 0.0310	0.4122 ± 0.0183	0.1247 ± 0.0459	0.0898 ± 0.0409	0.0846 ± 0.0118	0.6560 ± 0.0124

TABLE II. Average diffusivity, $\mathcal{D}$, with the variability indicated by the sample standard deviation, for each two–dimensional combination of asymmetric agents and obstacles considered. All results correspond to a 100×100 lattice and are calculated from an ensemble of $10,000$ identically prepared realizations.

C. Three–dimensional random walk with symmetric obstacles

We now demonstrate how to apply the algorithm to three–dimensional problems in which either $1 \times 1 \times 1$ or $2 \times 2 \times 2$ agents move on a lattice that is randomly populated with either $1 \times 1 \times 1$ or $2 \times 2 \times 2$ obstacles. To manage the computational expense of dealing with three–dimensional calculations we consider three–dimensional lattices that contain approximately the same number of lattice sites as those considered in the two–dimensional simulations in Sections III A and III B. We show a histogram of the ensemble calculations of $\mathcal{D}$ in Figure 6. The sample mean and variability, indicated by the sample standard deviation, are reported in Table III.

Similar to the results in two dimensions, we find that $\mathcal{D}$ decreases with ϕ for all combinations of agents and obstacles in three dimensions. Our calculations indicate that smaller $1 \times 1 \times 1$ obstacles are more effective at reducing $\mathcal{D}$ than larger $2 \times 2 \times 2$ obstacles, for both types of agents and for all obstacle densities considered. Furthermore, for all obstacle types and densities considered, these effects are most pronounced for larger $2 \times 2 \times 2$ agents than they are for smaller $1 \times 1 \times 1$ agents. These results are consistent with the general conclusions we reached for a two–dimensional systems in Section III A.

Similar to the two–dimensional results, we find that situations in which the agent and obstacle are the same size, the absolute size of the agent and obstacles relative to the lattice spacing affects $\mathcal{D}$. Generally, increasing the size of the agents and obstacles relative to the lattice spacing decreases $\mathcal{D}$. For example, for $\phi = 0.10$, we have $\mathcal{D} = 0.8463$ and $\mathcal{D} = 0.7136$ for systems involving $1 \times 1 \times 1$ and $2 \times 2 \times 2$ agents and obstacles, respectively.

Two sets of histograms of each ensemble of $\mathcal{D}$ are given in Figure 7, corresponding to two differently sized three–dimensional square lattices: $P = Q = R = 15$ and $P = Q = R = 25$. Our results indicate that increasing the size of the lattice does not alter the mean $\mathcal{D}$ for any combination of agents and obstacles considered. However, increasing the size of the lattice reduces the sample standard deviation, which is consistent with our exact calculations of $\mathcal{D}$ in two dimensions.

	$1 \times 1 \times 1$ Agent		$2 \times 2 \times 2$ Agent	
ϕ	$1 \times 1 \times 1$ Obstacles	$2 \times 2 \times 2$ Obstacles	$1 \times 1 \times 1$ Obstacles	$2 \times 2 \times 2$ Obstacles
0.05	0.9236 ± 0.0016	0.9487 ± 0.0021	0.6313 ± 0.0209	0.8558 ± 0.0111
0.10	0.8463 ± 0.0031	0.8962 ± 0.0042	0.3305 ± 0.0380	0.7136 ± 0.0215
0.15	0.7686 ± 0.0090	0.8426 ± 0.0061	0.1044 ± 0.0421	0.5742 ± 0.0308

TABLE III. Average diffusivity, $\mathcal{D}$, with the variability indicated by the sample standard deviation, for each three–dimensional combination of agents and obstacles considered. All results correspond to a $25 \times 25 \times 25$ lattice and are calculated from an ensemble of $10,000$ identically prepared realizations.

D. Higher values of obstacle density

We now consider a wider range of values of ϕ, $0 \leq \phi \leq 0.5$, for a situation in which 1×1 agents move on a lattice that is randomly populated with 1×1 obstacles. Without any obstacles, we have $\phi = 0$ and $\mathcal{D} = 1$. Additional results in Table IV and Figure 8 report values of $\mathcal{D}$ for $0 \leq \phi \leq 0.5$. As ϕ increases, $\mathcal{D}$ decreases, as expected. For $\phi = 0.45$ and $\phi = 0.50$ we find that $\mathcal{D} = 4.46 \times 10^{-5}$ and $\mathcal{D} = 5.37 \times 10^{-8}$ which both round to zero at four decimal places. We find that the standard deviation

of the distribution increases with ϕ, and decreases as the size of the lattice increases, as shown in Figure 9.

ϕ	1×1 Agents with 1×1 obstacles
0.05	0.8907 ± 0.0011
0.10	0.7765 ± 0.0022
0.15	0.6564 ± 0.0100
0.20	0.5297 ± 0.0178
0.25	0.3980 ± 0.0210
0.30	0.2617 ± 0.0226
0.35	0.1275 ± 0.0201
0.40	0.0203 ± 0.0110
0.45	0.0000 ± 0.0007
0.50	0.0000 ± 0.0000

TABLE IV. Average diffusivity, $\mathcal{D}$, with the variability indicated by the sample standard deviation, for each two–dimensional combination of symmetrical agents and symmetrical obstacles considered. All results correspond to a 100×100 lattice and are calculated from an ensemble of $10,000$ identically prepared realizations.

IV. DISCUSSION AND CONCLUSION

A routine aim of applying a random walk model to replicate observations made from cellular and molecular experiments is to provide an estimate of the Fickian diffusivity [6–8]. While these concepts are relatively straightforward in an *in vitro* experimental setting, applying a random walk model to an *in vivo* situation is complicated by the fact the motion of cells and molecules is hindered by the presence of obstacles of varying shapes and sizes [9–12]. Previous approaches for estimating the diffusivity from a random walk model in a crowded environment involve performing repeated stochastic random walk simulations and collecting MSD data [13–16]. Since agent trajectory data can be very noisy and estimates of diffusivity require long–time MSD data, it can be computationally demanding to estimate the Fickian diffusivity from repeated simulations of such models.

In this work we calculate D using a different approach that provides an exact result without the need for performing repeated stochastic simulations [26–28]. Our results provide insight into the details of how varying the shape and size of the obstacles affects the Fickian diffusivity. Furthermore, we are also able to explore how varying the shape and size of the motile agent affects the diffusivity. In summary, our results show that the diffusivity is strongly dependent on the details of the obstacle field

and is sensitive to the density of obstacles, the size of obstacles, the shape of obstacles and the size of the obstacles relative to the agent.

Although our approach for calculating $\mathcal{D}$ overcomes certain limitations of relying solely upon stochastic simulation data, our approach is subject to other limitations. For example, the exact approach for calculating $\mathcal{D}$ is relevant to the situation where the obstacles are stationary only. This is relevant for certain problems, whereas other previous studies have considered mobile obstacles for which the exact calculations are not relevant. While the results presented in this study are limited to four different types of two–dimensional obstacles and agents, and two different types of three–dimensional obstacles and agents, the approach outlined here can also be applied to other lattices, such as a triangular lattice, and other types of agents and obstacles of varying shapes and size.

Acknowledgments: We appreciate support from the Australian Research Council (FT130100148). We also acknowledge assistance and resources provided by the High Performance Computing and Research Support group at QUT. We thank the two anonymous referees for their helpful comments.

V. APPENDIX

We now calculate the long–time diffusivity in each direction for the lattices shown in Figure 4(b)–(c). For the lattice in Figure 4(b), when we bias motion in the positive x–direction, the transition matrix is given by

$$\mathbf{T} = \begin{pmatrix} \frac{3+\epsilon}{4} & \frac{1+\epsilon}{4} & 0 & 0 & 0 & 0 \\ \frac{1-\epsilon}{4} & 0 & \frac{1+\epsilon}{4} & 0 & \frac{1}{2} & 0 \\ 0 & \frac{1-\epsilon}{4} & \frac{1-\epsilon}{4} & 0 & 0 & \frac{1}{2} \\ 0 & 0 & 0 & \frac{3-\epsilon}{4} & 0 & \frac{1-\epsilon}{4} \\ 0 & \frac{1}{2} & 0 & 0 & \frac{1+\epsilon}{4} & \frac{1+\epsilon}{4} \\ 0 & 0 & \frac{1}{2} & \frac{1+\epsilon}{4} & \frac{1-\epsilon}{4} & 0 \end{pmatrix},$$

and the mean velocity vectors are given by

$$\mathbf{v}_I = \left(-\tfrac{1}{4}, 0, \tfrac{1}{4}, \tfrac{1}{4}, -\tfrac{1}{4}, 0\right)^{\mathrm{T}},$$
$$\mathbf{v}_\epsilon = \left(\tfrac{1}{4}, \tfrac{1}{2}, \tfrac{1}{4}, \tfrac{1}{4}, \tfrac{1}{4}, \tfrac{1}{2}\right)^{\mathrm{T}}.$$

Equations (7)–(8) gives

$$\mathbf{n}_I = \left(\tfrac{1}{6}, \tfrac{1}{6}, \tfrac{1}{6}, \tfrac{1}{6}, \tfrac{1}{6}, \tfrac{1}{6}\right)^{\mathrm{T}},$$
$$\mathbf{n}_\epsilon = \left(\tfrac{1}{2}, \tfrac{1}{6}, -\tfrac{1}{6}, -\tfrac{1}{2}, \tfrac{1}{6}, -\tfrac{1}{6}\right)^{\mathrm{T}},$$

from which we arrive at the diffusivity

$$\mathcal{D}_x = \mathbf{v}_\epsilon \cdot \mathbf{n}_I + \mathbf{v}_I \cdot \mathbf{n}_\epsilon = 0.$$

This result is expected as the location of the two 1×1 obstacles on this lattice mean that the displacement of the x–component of the trajectory is always confined by the obstacles and cannot increase indefinitely. When we bias the motion in the positive y–direction, the transition matrix is given by

$$\mathbf{T} = \begin{pmatrix} \frac{3}{4} & \frac{1}{4} & 0 & 0 & 0 & 0 \\ \frac{1}{4} & 0 & \frac{1}{4} & 0 & \frac{1}{2} & 0 \\ 0 & \frac{1}{4} & \frac{1}{4} & 0 & 0 & \frac{1}{2} \\ 0 & 0 & 0 & \frac{3}{4} & 0 & \frac{1}{4} \\ 0 & \frac{1}{2} & 0 & 0 & \frac{1}{4} & \frac{1}{4} \\ 0 & 0 & \frac{1}{2} & \frac{1}{4} & \frac{1}{4} & 0 \end{pmatrix},$$

and the mean velocity vectors are

$$\mathbf{v}_I = \left(0, 0, 0, 0, 0, 0\right)^{\mathrm{T}},$$
$$\mathbf{v}_\epsilon = \left(0, 0, \tfrac{1}{2}, \tfrac{1}{2}, \tfrac{1}{2}, \tfrac{1}{2}\right)^{\mathrm{T}}.$$

Using Eqs. (7)–(8), we have

$$\mathbf{n}_I = \left(\tfrac{1}{6}, \tfrac{1}{6}, \tfrac{1}{6}, \tfrac{1}{6}, \tfrac{1}{6}, \tfrac{1}{6}\right)^{\mathrm{T}},$$
$$\mathbf{n}_\epsilon = \left(0, 0, 0, 0, 0, 0\right)^{\mathrm{T}},$$

giving a diffusivity of

$$\mathcal{D}_y = \mathbf{v}_\epsilon \cdot \mathbf{n}_I + \mathbf{v}_I \cdot \mathbf{n}_\epsilon = \frac{1}{3}.$$

We now consider the three–dimensional lattice depicted in Figure 4(c). The algorithm is identical to the algorithm for a two–dimensional lattice except that each lattice site has a maximum of six nearest neighbors instead of four. Introducing a bias in the positive x–direction, the transition matrix for this lattice is given by

$$\mathbf{T} = \begin{pmatrix} \frac{1}{3} & \frac{1}{3} & 0 & \frac{1}{3} & 0 & 0 & 0 \\ \frac{1}{3} & 0 & \frac{1}{3} & 0 & \frac{1}{3} & 0 & 0 \\ 0 & \frac{1}{3} & \frac{1}{3} & 0 & 0 & 0 & \frac{1}{3} \\ \frac{1}{3} & 0 & 0 & 0 & \frac{1}{3} & \frac{1}{3} & 0 \\ 0 & \frac{1}{3} & 0 & \frac{1}{3} & 0 & 0 & \frac{1}{3} \\ 0 & 0 & 0 & \frac{1}{3} & 0 & \frac{1}{3} & \frac{1}{3} \\ 0 & 0 & \frac{1}{3} & 0 & \frac{1}{3} & \frac{1}{3} & 0 \end{pmatrix},$$

and the mean velocity vector is given by

$$\mathbf{v}_I = \Big(0,0,0,0,0,0,0\Big)^{\mathrm{T}},$$
$$\mathbf{v}_\epsilon = \Big(0,\tfrac{1}{3},\tfrac{1}{3},\tfrac{1}{3},\tfrac{1}{3},\tfrac{1}{3},\tfrac{1}{3}\Big)^{\mathrm{T}}.$$

Equations (7)–(8) give

$$\mathbf{n}_I = \Big(-\tfrac{1}{7},\tfrac{1}{7},\tfrac{1}{7},\tfrac{1}{7},\tfrac{1}{7},\tfrac{1}{7},\tfrac{1}{7}\Big)^{\mathrm{T}},$$
$$\mathbf{n}_\epsilon = \Big(0,0,0,0,0,0,0\Big)^{\mathrm{T}},$$

and the diffusivity is

$$\mathcal{D}_x = \mathbf{v}_\epsilon \cdot \mathbf{n}_I + \mathbf{v}_I \cdot \mathbf{n}_\epsilon = \frac{2}{7}.$$

Because the lattice in Figure 4(c) is symmetric in each direction, it follows that $\mathcal{D}_x = \mathcal{D}_y = \mathcal{D}_z = 2/7$, and this can be confirmed by repeating the calculations by biasing the motion in the positive or negative y and z directions, respectively.

[1] EA Codling, MJ Plank, S Benhamou. J R Soc Interface. **5** 813 (2008)

[2] BD Hughes. *Random Walks and Random Environments*. Volume 1 (Oxford University Press, 1995)

[3] CA Athale, A Dinarina, F Nedelec, E Karsenti. Phys Biol. **11** 016008 (2014)

[4] H Yin, X Wen, T Zhou. Phys Biol. **10** 056012 (2014)

[5] PR Taylor, RE Baker, CA Yates. Phys Biol. **12** 016006 (2015)

[6] AQ Cai, KA Landman, BD Hughes. J Theor Biol. **245** 576-594 (2007)

[7] KK Treloar, MJ Simpson, DLS McElwain, RE Baker. J Theor Biol. **356** 71-84 (2014)

[8] ST Johnston, MJ Simpson, DLS McElwain. J R Soc Interface. **11** 20140325 (2014)

[9] T Alarcón, HM Byrne, PK Maini. J Theor Biol. **225** 257 (2003)

[10] HM Young, AJ Bergner, MJ Simpson, SJ McKeown, MM Hao, CR Anderson, H Enomoto. BMC Biology **12** 23 (2014)

[11] K Murase, T Fujiwara, Y Umemura, K Suzuki, I Ryota, H Yamashita, M Saito, H Murakoshi, K Ritchie, A Kusumi. Biophys J. **86** 4075 (2004)

[12] T Ando, J Skolnick. PNAS. **107** 18457 (2010)

[13] MJ Saxton. Biophys J. **66** 394-401 (1994)

[14] MJ Saxton. Biophys J. **72** 1744-1753 (1997)

[15] A Wedemeier, T Zhang, H Merlitz, C-X Wu, J Langowski. J Chem Phys. **128** 155101 (2008)

[16] A Wedemeier, H Merlitz, J Langowski. Europhys Lett. **88** 38004 (2009)

[17] DS Dean, IT Drummond, RR Horgan, A Lefevre, J Phys A. **37** 10459 (2004)

[18] AB Harris, Y Meir, A Aharony, Phys Rev B. **36** 8752 (1987)

[19] A Giacometti, H Nakanishi, A Maritan, NH Fuchs, J Phys A. **25** L461 (1992)

[20] NH Fuchs, H Nakanishi, Phys Rev A. **43** 1721 (1991)

[21] A Giacometti, H Nakanishi, Phys Rev. E. **50** 1093 (1994)

[22] E Vilaseca, A Isvoran, S Madurga, I Pastor, JL Garcés, F. Mas. Phys Chem Chem Phys. **13** 7396 (2011)

[23] AJ Ellery, MJ Simpson, SW McCue, RE Baker. J Chem Phys. **140** 054108 (2014)

[24] SR McGuffee, AH Elcock. PLoS Comput Biol. **6** e10000694 (2010)

[25] LG Bowden, MJ Simpson, RE Baker. J R Soc Interface. **10** 20130630.

[26] J-F Mercier, GW Slater, HL Guo. J Chem Phys. **110** 6050 (1999)

[27] J-F Mercier, GW Slater. J Chem Phys. **110** 6057 (1999)

[28] J-F Mercier, GW Slater. J Chem Phys. **113** 9109 (2000)

[29] D Stauffer, A Aharony. *Introduction to Percolation Theory* (Taylor and Francis, 1992)

[30] D Chandler. *Introduction to Modern Statistical Mechanics* (Oxford University Press, Inc. 1987)

[31] DT Gillespie. J Phys Chem-US. **81** 2340 (1977)

[32] TM Liggett, *Interacting Particle Systems* (Springer, 2005)

[33] Y Saad, *Iterative Methods for Sparse Linear Systems* (SIAM, 2003)

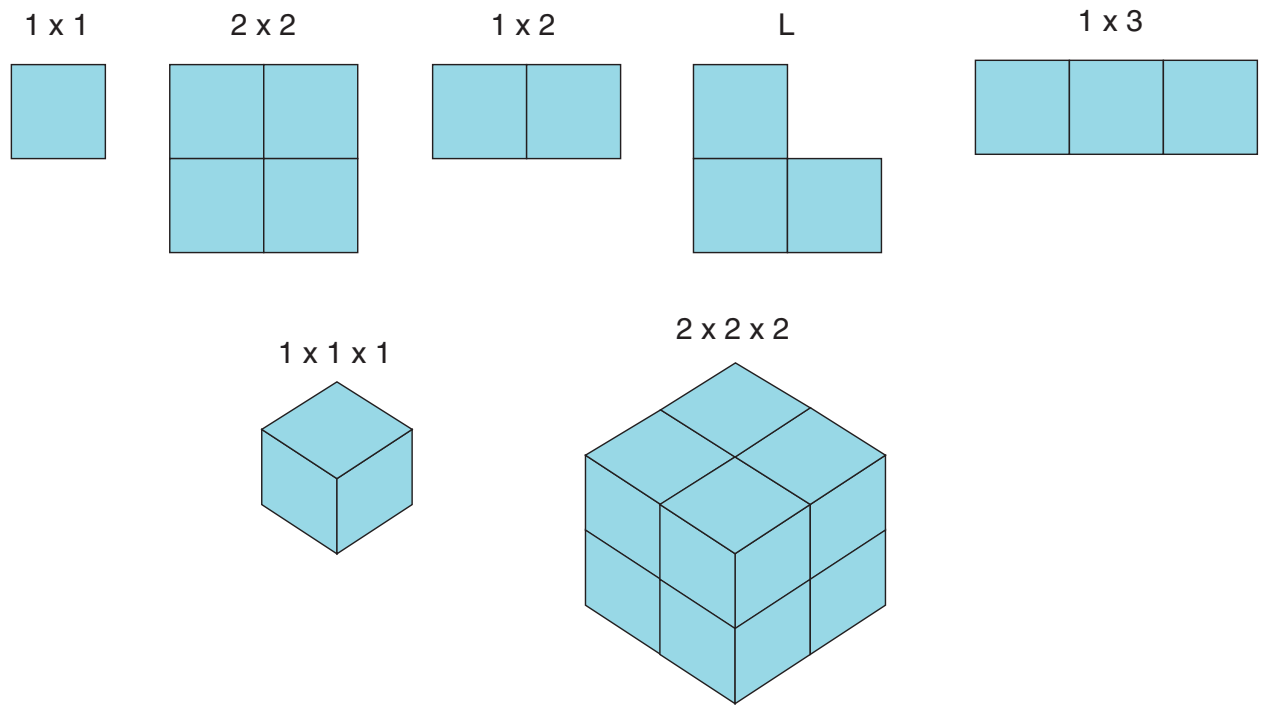

FIG. 1. Various sizes and shapes of the two– and three–dimensional agents and obstacles used in this study. The dimensions relative to the lattice spacing are indicated next to each object.

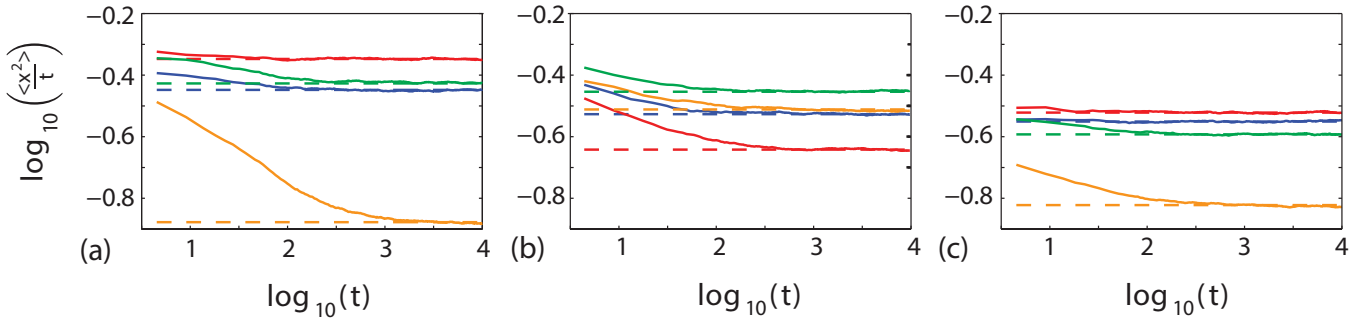

FIG. 2. Results in (a) show averaged MSD data, $\log_{10}(\langle x^2\rangle/t)$, for a 1×1 agent moving on a lattice populated with 1×1 (blue) and 2×2 obstacles (red), and similar data for a 2×2 agent moving on a lattice populated with 1×1 (orange) and 2×2 obstacles (green). Results in (b) show $\log_{10}(\langle x^2\rangle/t)$ for a 1×2 agent moving on a lattice populated with 1×1 obstacles (blue), a 1×1 agent moving on a lattice populated with 1×2 obstacles (red), an L–shaped agent moving on a lattice populated with 1×1 obstacles (orange) and a 1×1 agent moving on a lattice populated with L–shaped obstacles (green). Results in (c) show $\log_{10}(\langle x^2\rangle/t)$ for a $1\times1\times1$ agent moving on a three–dimensional lattice populated with $1\times1\times1$ obstacles (blue) and $2\times2\times2$ obstacles (red), and similar data for a $2\times2\times2$ agent moving on a lattice populated with $1\times1\times1$ obstacles (orange) and $2\times2\times2$ obstacles (green). The size of the lattice and density of obstacles are $M=N=10$, $P=Q=R=10$ and $\phi=0.1$, respectively. All random walk simulations are averaged over $50,000$ identically prepared realizations. We also superimpose, for each case, in each subfigure, a dashed horizontal line, $\log_{10}(2D_x)$, where D_x is the long–time Fickian diffusivity in the x–direction. Exact values of D_x are calculated using the Mercier–Slater algorithm using GMRES with a strict error tolerance 10^{-8}.

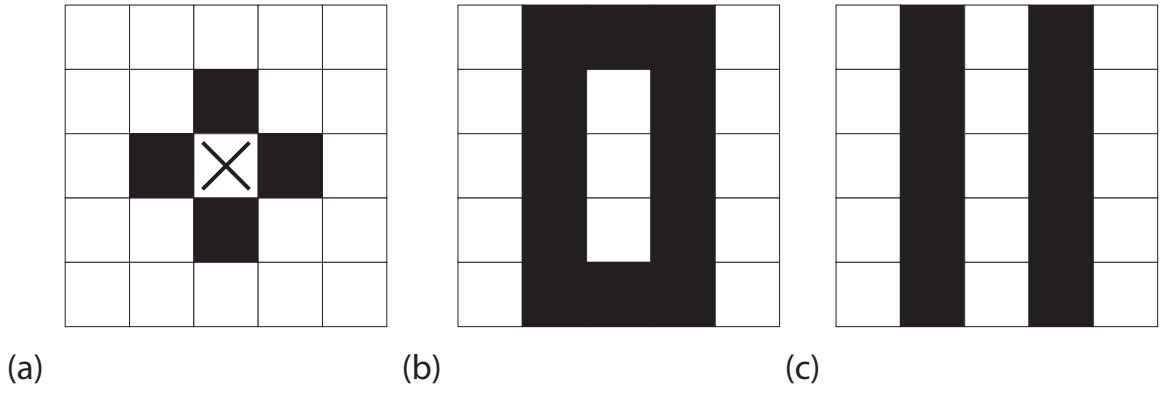

FIG. 3. (a) Special case (i), a lattice in which the obstacles (solid black) are laid down in a configuration so that the agent (represented by X) cannot move. (b)–(c) Special case (ii), two examples of lattices in which the obstacles (solid black) are laid down so that the lattice contains two or more independent closed regions.

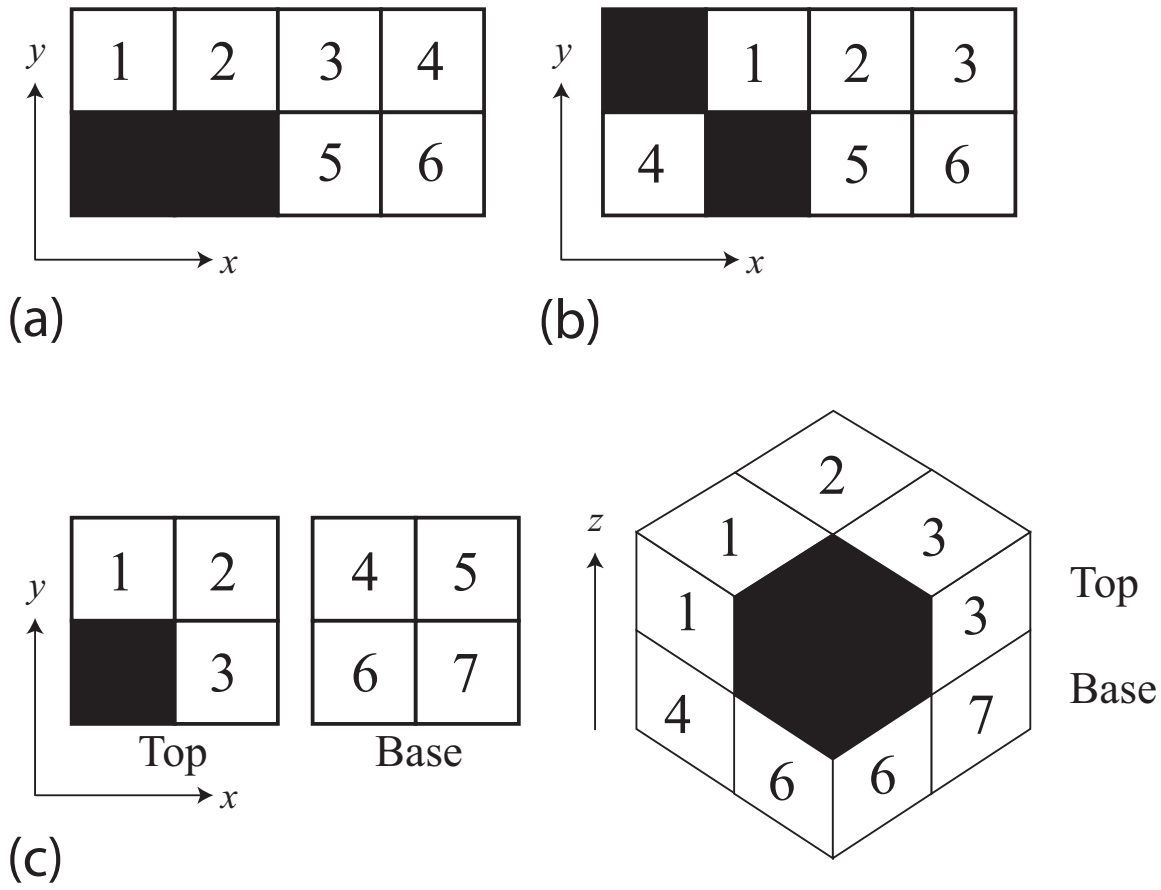

FIG. 4. (a) and (b) show two examples of two–dimensional lattices, used for the worked calculations. In each case the site numbering and obstacle locations (black) are given. (c) shows an example of a three–dimensional lattice, used for the worked calculations. The site numbering and obstacle location (black) is given. For the three–dimensional lattice we show both the entire lattice (lower right) as well as the two layers of the lattice (lower left).

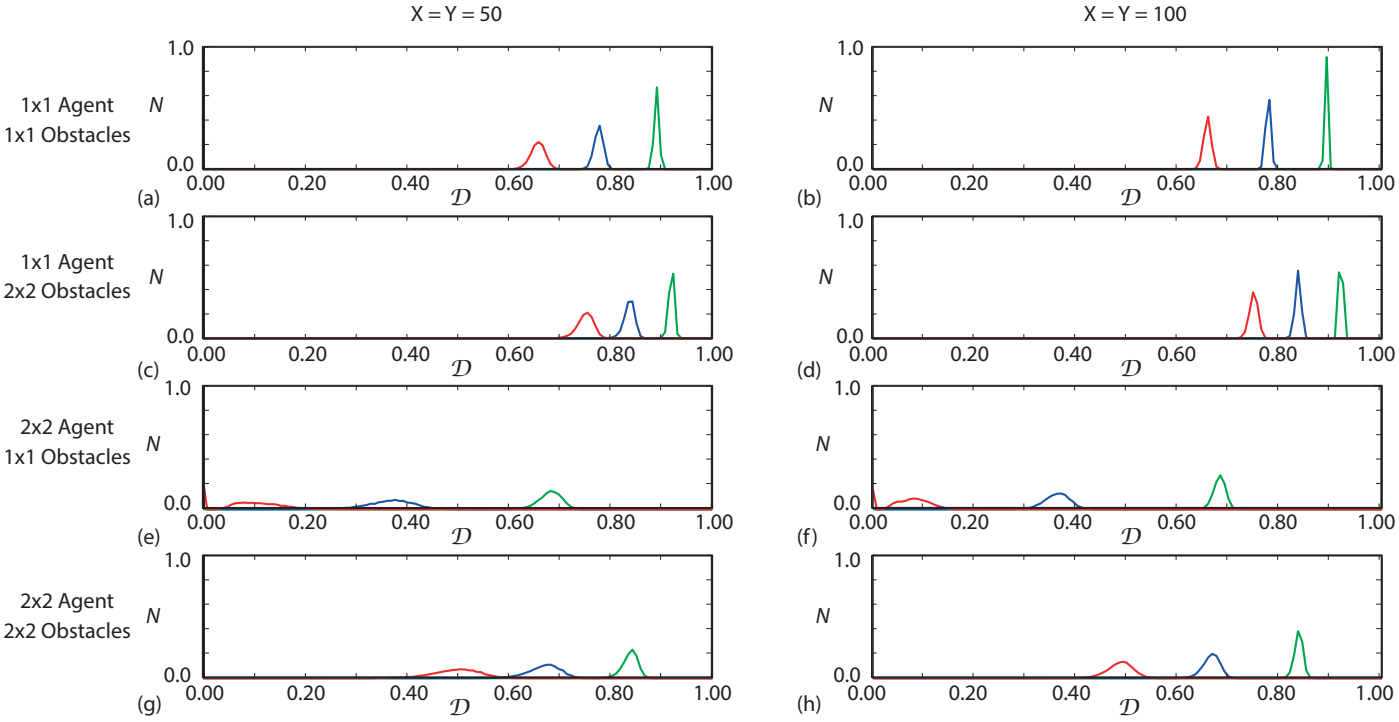

FIG. 5. Results in (a)–(h) show a normalized histogram of $\mathcal{D}$ values for 10,000 lattices which are randomly populated with obstacles to densities $\phi = 0.15$ (red), $\phi = 0.10$ (blue) and $\phi = 0.05$ (green). N is the normalised number of counts in each histogram box. Results in (a)–(d) and (e)–(h) correspond to 1×1 and 2×2 agents, respectively. Results in (a)–(b); (e)–(f) and (c)–(d); (g)–(h) correspond to 1×1 and 2×2 obstacles, respectively. Results in (a), (c), (e), (g) and (b), (d), (f), (h) correspond to $M = 50$ and $M = 100$, respectively. All histograms are constructed with 250 equally spaced intervals between 0.00 to 1.00. The solution of all systems of linear equations use GMRES with a strict error tolerance 10^{-8}.

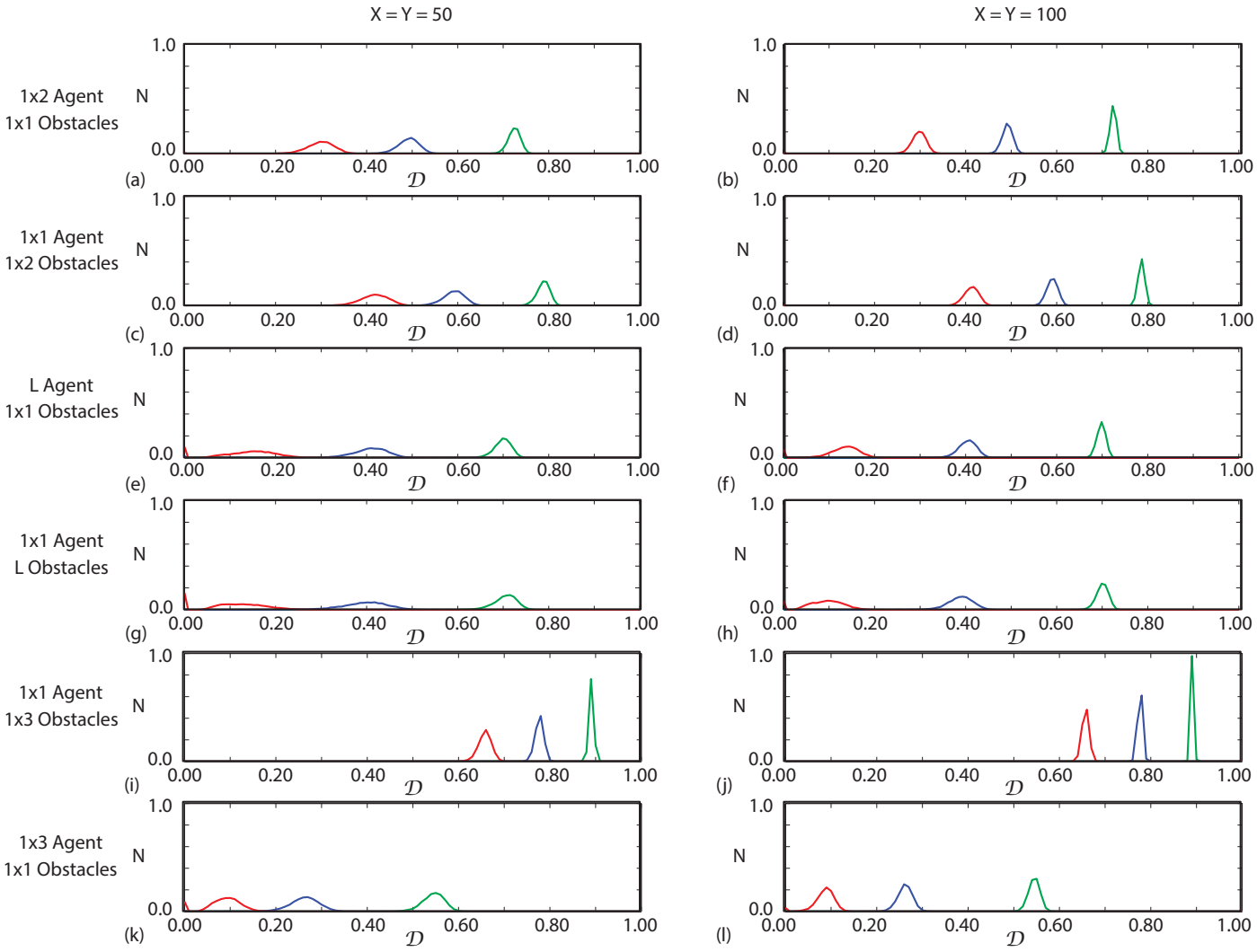

FIG. 6. Results in (a)–(l) show a normalized histogram of $\mathcal{D}$ values for 10,000 lattices which are randomly populated with obstacles to densities $\phi = 0.15$ (red), $\phi = 0.10$ (blue) and $\phi = 0.05$ (green). N is the normalised number of counts in each histogram box. Results in (a)–(b), (c)–(d), (e)–(f), (g)–(h), (i)–(j), (k)–(l) correspond to a 1×2 agent with 1×1 obstacles, a 1×1 agent with 1×2 obstacles, an L–shaped agent with 1×1 obstacles, a 1×1 agent with L–shaped obstacles, a 1×1 agent with 1×3 obstacles, and a 1×3 agent with 1×1 obstacles, respectively. Results in (a), (c), (e), (g), (i), (k) and (b), (d), (f), (h), (j), (l) correspond to $X = Y = 50$ and $X = Y = 100$, respectively. All histograms are constructed with 250 equally spaced intervals between 0.00 to 1.00. The solution of all systems of linear equations use GMRES with a strict error tolerance 10^{-8}.

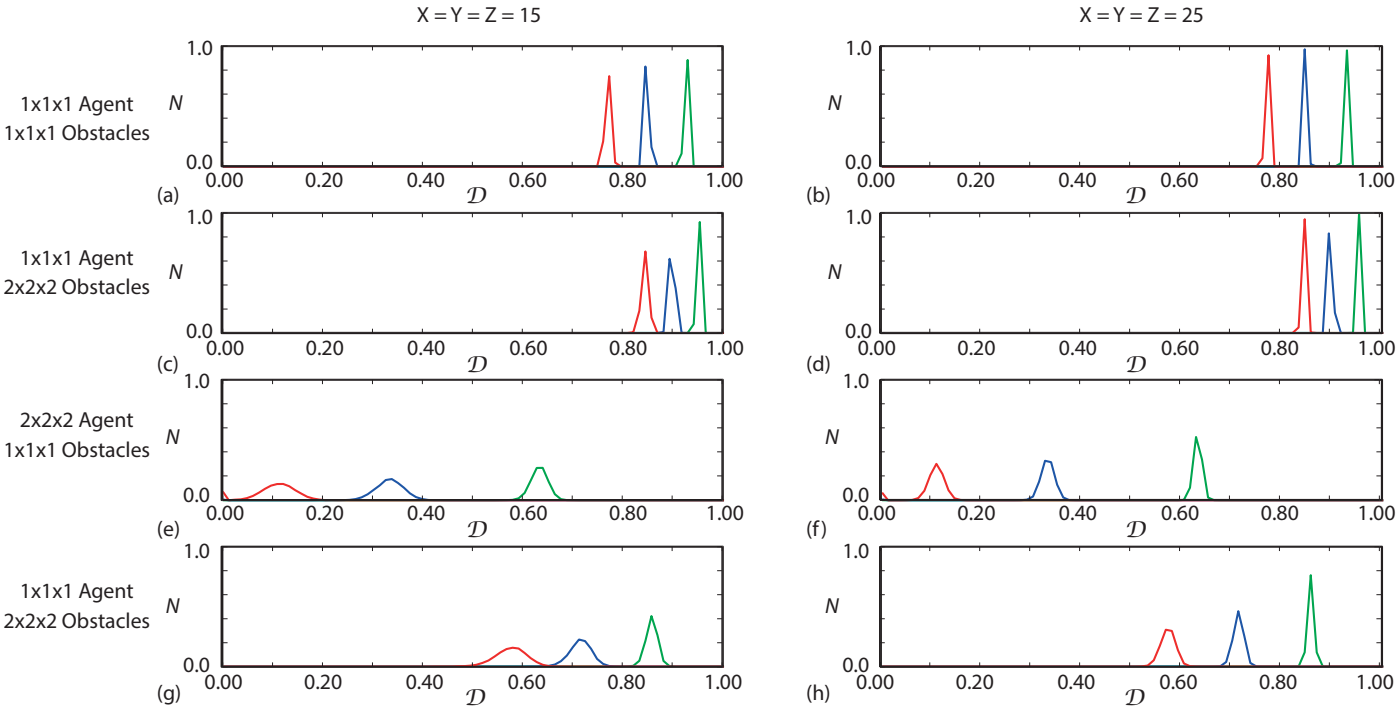

FIG. 7. Results in (a)–(h) show a normalized histogram of $\mathcal{D}$ values for 10,000 lattices which are randomly populated with obstacles to densities $\phi = 0.15$ (red), $\phi = 0.10$ (blue) and $\phi = 0.05$ (green). N is the normalised number of counts in each histogram box. Results in (a)–(b), (c)–(d), (e)–(f) and (g)–(h) correspond to simulations involving a $1 \times 1 \times 1$ agent with $1 \times 1 \times 1$ obstacles, a $1 \times 1 \times 1$ agent with $2 \times 2 \times 2$ obstacles, a $2 \times 2 \times 2$ agent with $1 \times 1 \times 1$ obstacles, and a $2 \times 2 \times 2$ agent with $2 \times 2 \times 2$ shaped obstacles, respectively. Results in (a), (c), (e), (g), and (b), (d), (f), (h) correspond to $P = Q = R = 15$ and $P = Q = R = 25$, respectively. All histograms are constructed with 250 equally spaced intervals between 0.00 to 1.00. The solution of all systems of linear equations use GMRES with a strict error tolerance 10^{-8}.

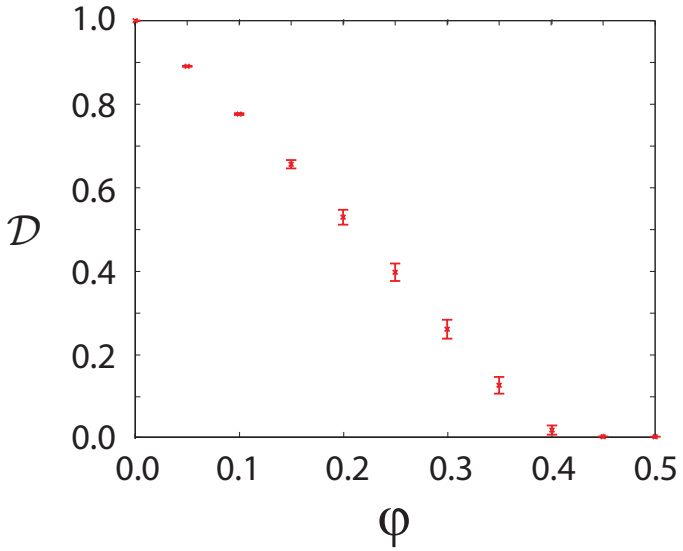

FIG. 8. $\mathcal{D}$ for a 1×1 agent moving on a 100×100 two–dimensional lattice randomly populated with 1×1 obstacles to density ϕ. Results are given for $0 \leq \phi \leq 0.5$. Here the data points correspond to $\langle \mathcal{D} \rangle$, constructed using 10,000 identically prepared lattices. The error bars denote the sample standard deviation.

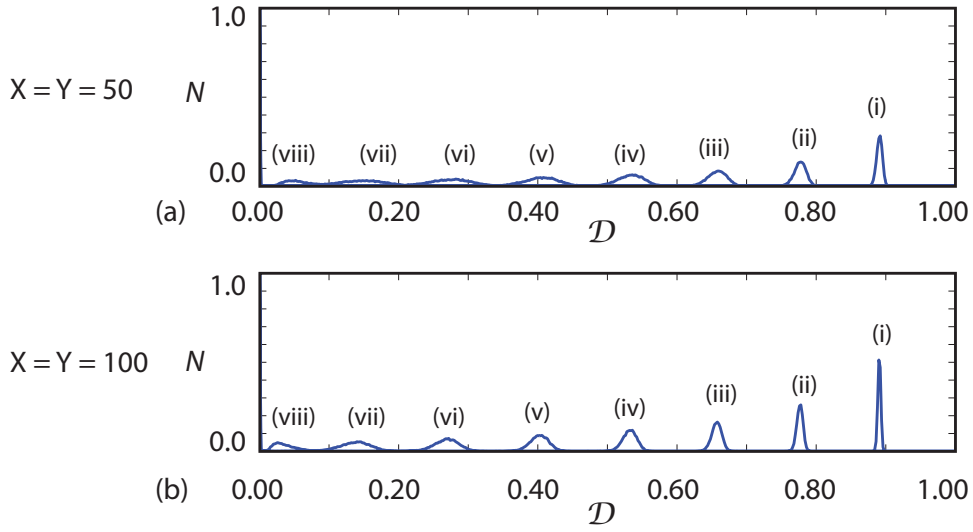

FIG. 9. Results in (a)–(b) show a normalized histogram of $\mathcal{D}$ values for $10,000$ lattices which are randomly populated with obstacles to densities $\phi = 0.05, 0.10, 0.15, 0.20, 0.25, 0.30, 0.35$ and 0.40, which are labeled in sequential order (i)–(viii), respectively. N is the normalised number of counts in each histogram box. Results in (a)–(b) correspond to $X = Y = 50$ and $X = Y = 100$, respectively. All histograms are constructed with 250 equally spaced intervals between 0.00 to 1.00. The solution of all systems of linear equations use GMRES with a strict error tolerance 10^{-8}.